A Sketch-Based Interface for Annotation of 3D Brain Vascular Reconstructions

David Selosse, Jérémie Dequidt, Laurent Grisoni

A Sketch-Based Interface for Annotation of 3D Brain Vascular Reconstructions

David Selosse, Jeremie Dequidt, Laurent Grisoni

A Sketch-Based Interface for Annotation of 3D Brain Vascular Reconstructions

David Selosse[*][†], Jeremie Dequidt[‡][†], Laurent Grisoni[§][†]

Project-Teams MINT

Abstract: Within the medical imaging community, 3D models of anatomical structures are now widely used in order to establish more accurate diagnoses than those based on 2D images. Many research works focus on an automatic process to build such 3D models. However automatic reconstruction induces many artifacts if the anatomical structure exhibits tortuous and thin parts (such as vascular networks) and the correction of these artifacts involves 3D-modeling skills and times that radiologists do not have. This article presents a semi-automatic approach to build a correct topology of vascular networks from 3D medical images. The user interface is based on sketching; user strokes both defines a command and the part of geometry where the command is applied to. Moreover the user-gesture speed is taken into account to adjust the command: a slow and precise gesture will correct a local part of the topology while a fast gesture will correct a larger part of the topology. Our system relies on an automatic segmentation that provides a initial guess that the user can interactively modify using the proposed set of commands. This allows to correct the anatomical aberrations or ambiguities that appear on the segmented model in a few strokes.

Key-words: Sketching,3D Reconstruction, Model Annotation

* e-mail: david.selosse@lifl.fr
† LIFL, CNRS/IRCICA - INRIA Lille-Nord-Europe
‡ e-mail: jeremie.dequidt@lifl.fr
§ e-mail: laurent.grisoni@lifl.fr

Une interface Sketch-Based pour l'Annotation de Reconstruction 3D Vasculaire Cérébrale

Résumé :   Dans le domaine de l'imagerie médicale, la modélisation 3D de structures anatomiques est maintenant largement utilisée dans l'optique d'établir des diagnostics plus précis qu'avec des données basées sur des images 2D. Aujourd'hui, de nombreux travaux mettent l'accent sur les méthodes automatique de reconstruction de modèles 3D mais ces méthodes induisent de nombreuses erreurs. Avec une structure anatomique (réseau cérébral) présente des parties assez fines et tortueuses, des erreurs sont introduites, cela nécessitent de la correction du modèle 3D, mais aussi des compétences et des heures que les radiologistes ne possèdent pas. Cet article présente une approche semi-automatique de reconstruction d'une topologie correcte de réseaux vasculaires issus d'images médicales en 3D. Notre système repose sur une segmentation automatique qui fournit une estimation initiale dont l'utilisateur peut modifier interactivement en utilisant un jeu proposé de commandes basées sur le croquis. Cela permet de corriger les aberrations anatomiques ou les ambiguÃ¯tés qui apparaissent sur le modèle segmenté en quelques traits.

Mots-clés :   Annotation de modele, Reconstruction 3D, Sketching

1   Introduction

Medical Imaging has been the major improvement in medicine over the last decades as it allows non-invasive accurate diagnoses of pathologies and diseases. The better the diagnosis, the more efficient is the medical procedure. To do so, acquisition devices such as MRI or CT scanners are able to provide high-resolution images of the human body that can be combined in order to build 3D models of anatomical structures (the reader may refer to [4] for a complete overview of 3D reconstruction techniques).

The process of building 3D model given 2D slices involves a pipeline of many complex steps like noise and bias removal, thresholding, segmentation. . . This process is fully automatic or requires very few human intervention (possibly the selection of areas of interest or the threshold definition) to be easily used in a pre-operative context by radiologists. However under some circumstances, theses techniques fail to provide a usable 3D model. For instance, within the interventional radiology context, the limited resolution of acquisition devices combined with small, thin, tortuous blood vessels make it very difficult to achieve an accurate reconstruction of the vascular network. Indeed artifacts such as holes, unwanted blending, very noisy surfaces are common on such reconstructed models (see figure 1). Usual and efficient solutions such as a combination of a statistical atlas [12] and non-rigid deformation on medical images in order to avoid these artifacts are not convenient here since vascular networks exhibit high individual variability. In practice, experienced radiologists can interpret these 3D models and *mentally deduce* correct reconstructions. But they do not have the skills in 3D modeling, nor time, to refine the raw data and perform a valid 3D model, that would be understandable by interns or radiology beginners.

In this article we propose a sketch-based interface that would allow experienced radiologists to *annotate* models, so that beginners would more easily understand 3D reconstructions. In that context, annotating geometry means associating some arbitrary part of geometry to some predefined arbitrary tag. Tags may be further used to exhibit semantic informations such as reconstruction errors, potential pathologies, non-consistent local topologies, or local topological informations (vessel segments, Y-structures identifications). The latter is a critical question about brain vascular system, because it extends the potential use of geometry annotation systems: such a functionality may significantly simplify and speed-up the task of defining the vessel network, and thus open the way to patient-specific simulation of catheter insertion, for instance [2].

The contribution of this article is a set of sketch-based commands (and associated algorithmic tools) that, all together, provide an intuitive way to structure existing 3D models of blood vessel. The ouput of our system is the same geometry as initial, mesh triangles being tagged according to the desired annotation. The main originality of the proposed system is to take into account the main purpose of vascular network reconstruction, i.e. diagnosis. Our system can be used both for specifying vessel network skeleton, and also efficiently tag potentially pathologic areas in the reconstruction (in this work we restrict to aneurysms and unwanted branchings). Natural interaction modality is given in order to make our tool usable without requiring skills in modeling. This paper focuses on the topology of the 3D data as correct topology of patient-specific vessel networks may be used to guide the 3D reconstruction.

On most systems, user strokes are more and more used following two different ways to interpret them: first, as a set of points, a curve that can be used as a basis for some numerical process (fitting, optimization, etc...); second, a shape that is associated to some recognizing system, and allows application to interpret command (see related works, section 2). To our knowledge, no

existing system takes advantage of user gesture using these two views at the same time. In this work we consider user strokes both as command that are to be recognized, *and*, anywhere relevant, a curve that has a meaningful position relatively to the reconstructed geometry. The main advantage of our approach is that each command also has context that can be used for selection of geometry, without having user to specifically select each triangle.

Practically speaking, our contribution is to propose three following categories of strokes, aimed at topology edition of brain vascular systems. These three kind of strokes can be applied on a 3D reconstruction mesh. The algorithm employed for stroke recognition is very straightforward; however we introduce a original way to combine strokes and the way we use the stroke position to execute a command is novel.

- line segment: creation of a topological element (tubular section): user stroke is used both as a command, and as a position input for the geometry to be associated with this topological segment. A propagation algorithm is used to determine the geometry that is to be associated. This is described in section 4

- Circular stroke: depending on what is actually inside the convex hull of the stroke, two different actions are proposed:

    - When two or more topological segment are inside the convex hull of the stroke, the command associates together the topological elements (local tree structure). This is the main command for vascular tree structuration (see section 5).

    - When the convex hull of the stroke covers both some existing topological segment and non-connex geometry, the command is interpreted as linking geometry to an existing topological element: in our case, such a functionality may be needed when explicitly associate potentially non-connected geometry to some previously defined topological element.

- Walking stick (flipped "L" shape): this command is interpreted as dissociate geometry from a topological element: this is the command that is available for particularizing the geometry that is potentially significant of some aneurysm. For such an operation, we provide user with the ability to efficiently specify the part of geometry that is to be marked as "pathologic", by comparison to automatic approaches. It is detailed in section 6.

The contributions of this article also lie in the fact that the *position* of gesture command on the screen is important: as will be explained further, gesture location impacts the part of reconstructed geometry that is involved, or the topological elements to be connected. We think this idea can be seen as a specialized way to propose parameterized gesture [20], which is a convenient way for the user to specify informations on geometry, in cases where user can easily determine information that would be difficult to extract automatically.

2   Related Works

A lot of work study automatic reconstruction of surfaces from medical data: a complete overview of this point would go, alone, far beyond the scope of this paper. Different workflows exist to produce a smooth surface based on voxelized datasets, each of these workflow relies on different steps such as thresholding, active surfaces [9]... Despite recent improvements in image processing and reconstruction, reconstruction of vascular structures still requires human knowledge and interaction [11, 24] or strong assumptions [5] on the structures to segment to produce a relevant and accurate 3D model.

Few works use an interface-based approach for 3D model topological reconstruction. [8] proposes a method for creating 3D models topology based on sketching (see [13, 15] for a general introduction on using sketch for 3D modeling). sketch can also be used as a basis for the generation and control of realistic physically-based generation [21]: in such a work, both geometry and mechanical properties of hair is defined using sketch. [18] proposes a sketch-based approach that combines surface reconstruction with user dynamic feedback, through ad-hoc interface, on reconstruction errors. This approach takes advantage of volumetric field analysis, in order to adapt topology dynamically [6]. [17] proposes a specific method for automatic reconstruction of 3D models topology. [3] describes a method for automatically extracting topological structure for simple configurations. We refer the reader to [7] for a survey on sketch-based methods for topology modification of meshes. [19] describes how to extract skeleton curve from point cloud, possibly with large parts missing. [14] proposes a sketch-based interface for geometry deformation. [23] uses sketch-based interface for piecewise parameterization of meshes. Finally, let us also mention that sketch-based interfaces are also studied in the human computer-interaction field [10].

In the specific case of 1D structures, Pihuit & al [16] describe a system that interactively construct 3D models of vascular system by taking advantage of drawing conventions of medical community. [1] describes a method for interactive skeleton creation from intensity volumes. This approach efficiently extracts skeleton, from user input, and is also able to create branches.

A lot of work has been achieved on the recognition of shapes for command. [22] recently described a nice and reasonnably well working method for curve classification.

3 Overview of the approach

The major concern of the proposed approach is to guarantee intuitive yet fast manipulation of reconstructed surfaces. The work described in the following sections allows the user to edit the result of an automatic reconstruction (ambiguity removal, topology modification...) in order to provide a correct topology which is mandatory for an accurate diagnosis. To achieve theses goals, our approach relies on local and simple algorithms close to sketching algorithms to produce an accurate topology of 1D structures.

The prerequisite to our approach is a 3D model of a vascular structure possibly made of hundreds of small vessels. Our method is based on triangular mesh structure because it allows interactive rendering and editing even with thousands of triangles. Volumetric data or implicit surfaces are too cost-intensive in resources terms [6] and require significant processing time to display or modify a part of the model. Our triangular mesh is generated using rotational angiography images (3DRA) and standard thresholding on images, marching cubes and an active surface algorithm to produce a smooth surface of the considered structure. The purpose of the surface extraction is twofold: first, it makes it easy to visualize the overall shape of the anatomical structure and second, it serves as a support for the computations. The reader may refer to [9] for further details on automatic reconstruction and active surfaces.

The topology editing process is achieved by user gestures and each single gesture contains a command and locality information to edit the geometry. Extracting command from human gesture requires some algorithms of gesture recognition that interpret the gesture from a set route. This set consists of all possible and useful commands for the annotation model. The algorithm

"$1 dollar Recognizer" [22] is simple and, in our case very easily filled the criteria of speed in a context of interaction. Our use is that each user stroke will be compared to predefined shapes (which can be defined by learning or by very specific forms).A propability calculation is made between each stroke and predefined shape. If one of the recognizer template is almost similar to the user shape, the calculation's result is closer to 1. In our context: 1 form = 1 order, we also note that even this type of recognition is robust, the base forms require not to be topologically close.

The second step of our topology editing algorithm is to express the 2D stroke in 3D using a picking algorithm. It gives us the triangles that have been crossed by the stroke. However, as we want our approach to be robust even if deployed on devices where gesture can be very approximate (such as touch tables for instance), the picking algorithm has to be improved to handle inaccurate stroke. Indeed the discrete points that describes the user's stroke may not touch the surface and then do not pick triangles of the surface. To solve this problem, points with missing z-coordinate (e. g. that have not crossed the surface and therefore do not have a correct depth) get an average depth based on the depth of their neighbors. The 3D stroke has to be paired with the part of the surface the user want to edit. This part is the most time-consuming part of the overall approach. In order to make it fast enough for interactive applications, a combination of spatial hashing and winged-edge structure is implemented. Using this pre-computed topology of the surface, a propagation starts from the triangles hit by the picking algorithm in order to select every triangle close to the stroke. The stroke is then approximated as a straight line segment, and each triangle of the surface and each point of the stroke is projected on this line segment. Projected items that are closed are finally paired together.

The following sections describe the 3 gestures (see in 2) that the user can perform on the geometry to edit it: first, the linear stroke to extract a tubular section of the geometry; second, the circular stroke that connects two or more parts of the geometry and third, the walking stick stroke to topologically disconnect two parts of the geometry.

4   Linear Stroke

Our system presents two different ways to use linear strokes that only differs by their propagation on the mesh: *user-specified propagation* limits the topological element to the bounds of the user stroke, and *automatic propagation* propagates the topological vessel as much as possible, i.e. until bounds of reconstruction, or Y-structures connecting vessels together. We discriminate between the two sub-types of command using user gesture average speed, stating that accurate stroke is classically achieved slower than coarse ones. This empirical statement may need to be scientifically verified, but is very simple to implement, and appears to provide good results in our tests.

To define the skeleton, we use a few steps to define the seed propagation (see figure 3):

- Step 1: Select the entry point of interaction on the model by calculating dot product of picked triangle normals (resulting from the interaction) and the camera direction. We select the input point witch maximizes this product.

- Step 2: Starting from defined input point, we create two points on model by front / back picking and calculate the center between these two points.

- Step 3: Creating a ring of points (with arbitray step) around an axis through the center point and collinear with the general direction (in orange).

- Step 4: Recovery of triangles segment, starting from picking the front and rear, close to the ring created by comparing the distance between each point (according to an arbitrary threshold). Thus a seed (initial ring) is created allowing the propagation of ring ring for the purpose of creating local topology.

The propagation algorithm uses a stack by two initially filled seeds and each element of which bears the direction of propagation, the first seed uses the direction of propagation identical to that of the interaction, and the other the opposite direction. As long as the stack is not empty, the current element is treated in the popping, and calculating the next ring (depending on the direction of propagation), selecting the neighboring triangles to the current ring to create a new closed ring. Then propagation is fired again along the user gesture to extand the current skeleton and add other rings to the tagged one. Once rings are added, their center is computed and a curved is computed linking the center of the rings, therefore creating the medial axis of the selected vessel. (see figure 4: a and b).

Stopping the propagation depends on the sub-type of command that is achieved. The user-specified propagation is straightforward, as all user interaction points define the range of geometry that has to be structured. The automatic propagation approach is a bit more involving: to obtain a spread in the entire segment, it is necessary to define a proper stop criteria for propagation criterion based on triangles points which compose the current ring, the center of this ring is used as a seed for the skeleton (see figure 4: c) This criterion we propose is defined this way:

- Step1: We calculate the barycenter of the current ring using vertices of local mesh.

- Step2: We calculate vectors start from barycenter to each triangle center (in orange on in figure 5), at this point we have the same number of vectors that normals.

- Step3: We calculate one by one the scalar product of one vector and the associated normal (in red in in figure 5), and we made a mean of these scalar products to provide an evaluation of the geometric "opening" (step 3, case 2 in figure 5) or "closing" (step3, case 1 in figure 5) of the structure, while standard deviation provides an evaluator of irregularity of it (possibly the presence of reconstruction of artifacts). We use scalar product means and standard deviation, along with some basic thresholding technique for decision of stopping propagation.

5   Circular Stroke

With the first gesture, the user defines a local topology through strokes. With our processing, the local topology is associated with the 3D-model by tagging triangles to form a local skeleton. The association between a stroke and relevant triangles of the 3D-model is mandatory to ensure the topological reconnections of the local skeletons. Once the user gesture has been identified as a circular gesture, an algorithm identifies, among the triangles picked by the gesture (see figure ??), the ones that are already tagged. Given the intricacy of brain veins and arteries and the possible inaccuracy of the user-gesture, control must be added to prevent the connection of distant (from a geodesic or topological point of view) skeleton parts. To do so, our strategy is to only connect skeleton parts where one of their extremities is enclosed by the circular gesture. This process is achieved using the parametric abscissa of each triangle. As each topological segment is a 3D curve and because each triangle is attached to a single skeleton part, we can compute the parametric abscissa of the projected triangle on the curve. Then topological segments are connected if and only if they have some triangles picked AND the picked triangles have their

parametric abscissa lesser than 0.1 or greater than 0.9. This arbitrary threshold enables us to prevent connecting topological segments whose extremities may not be joined.

6   Walking Stick Stroke

The main issue raised with automatic reconstruction of medical images is the presence of un-wanted blending that may merge two vessels or a vessel and an aneurysm (the so-called *kissing vessels* artifact). The purpose of the walking stick stroke is to handle this issue by selectively add local geometry to the reconstructed topology.

The main idea is, in a first step, to sketch the wanted topology directly on the model. The gesture, which seems to be a stroke, need to starts from an already tagged segment (with an existing skeleton) in order to extend the existing skeleton (see the upper part of figure 6). In a second step, the user must end his gesture by a kind of break command (direction change and/or more accurate gesture) in order to differentiate this gesture with stroke one. We named the first part of the gesture the *command* and the second part the *parameter*, all this steps are describe in figure 6. We perform an analysis of the gesture to "split" the curve of the user in two, by detecting the maximum curvature of the stroke using:

$$c(t) = \frac{x'(t)y''(t) - y'(t)x''(t)}{(x'(t)^2 + y'(t)^2)^{3/2}}$$

Once the walking stick stroke has been characterized, we need to compute the topological elements induced by this gesture.The aim is to discriminate some parts of the geometries that are identified as merged elements (artifacts). As explained previously, the gesture should start from a already tagged part to known the distance between skeleton and the vessel surface, and the vessel's direction to apply this characteristics to the new topology, considering that we considerthe that distance and direction vary smoothly on merged structure.

We can combine the *command* curve drawn by the user and this additional data to build an extended topological element and tag triangles (with picking) under the *command* gesture part. The *parameter* curve allows to discard which part of the merged structure are identified by the user as non relevant. The figure 7 illustrates the walking stick stroke gesture on model with the initial step .

A C++ application has been developed to illustrate and benchmark the various propositions in this paper. The application allows the user define the topology using the three strokes available. This application has been tested using both synthetic and patient-specific data and allows in few strokes to define the correct topology around an area of interest.

Some performance measurements were made on different models in order to estimate the computation time of the various gestures and propagation modes. These measurements was made on HP Z400 Workstation: 4.0Go RAM, Quad-Core Intel XEON W3520 at 2.66Ghz and NVidia Quadro FX580 for graphics rendering. The models tested are: a simple tubular model (with a large section), a model Y generated (ordered mesh), a local part of a patient brain vascular network, and the full vascular network.

Table 1 highlights the impact of the model's type used in the processing time of a linear stroke interaction with an automatic propagation, using a stopping criterion set at 0.50 (see section 4). This is the most consuming time part of the algorithm and varies almost linearly with respect to the number of triangles.

Model Type: facets	Inputs	Treated	Time
Simple: 4384	7	3630	2386 ms
Y Topology: 3136	7	758	515 ms
Patient part: 8603	6	3955	1966 ms
Patient full: 45496	5	4926	2746 ms

Table 1: Measured performance on several test cases (full propagation) with the same number of inputs

7   Conclusion and Future Work

The paper presents three different gestures that allow to define the topology of a vascular network. The main originality of our approach is to handle the user gesture as both a command and data to apply the command. Future works will focus on the validation of the three gestures as some arbitrary threshold have been set. Moreover, a clinical validation is required in order to asses the relevance of our method compared to the clinicians needs. As a consequence, additional features may be derived from this topology definition such as local geometry reconstruction and / or modification.

References

[1] S. S. Abeysinghe and T. Ju. Interactive skeletonization of intensity volumes. *Vis. Comput.*, 25:627–635, April 2009.

[2] J. Dequidt, M. Marchal, C. Duriez, E. Kerrien, and S. Cotin. Interactive Simulation of Embolization Coils: Modeling and Experimental Validation. In *Medical Imaging Computing and Computer Assisted Intervention -MICCAI'08*, volume 5241 of *Lecture Notes in Computer Science*, pages 695–702, New York City, United States, 2008. Springer. The original publication is available at www.springerlink.com INRIA grant (ARC Simple).

[3] J. Egger, Z. Mostarkic, S. Grosskopf, and B. Freisleben. A fast vessel centerline extraction algorithm for catheter simulation. In *ISBMS '07: Proceedings of the Twentieth IEEE International Symposium on Computer-Based Medical Systems*, pages 177–182, Washington, DC, USA, 2007. IEEE Computer Society.

[4] A. El-Baz, R. Acharya, M. Mirmehdi, and J. Suri, editors. Springer, March 2011.

[5] O. Friman, M. Hindennach, and H.-O. Peitgen. Template-based multiple hypotheses tracking of small vessels. In *Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, pages 1047 –1050, may 2008.

[6] J. C. Hart, A. Durr, and D. Harsh. Critical points of polynomial metaballs. In *Implicit Surface?98 Proc*, pages 69–76, 1998.

[7] T. Ju. Fixing geometric errors on polygonal models: a survey. *J. Comput. Sci. Technol.*, 24:19–29, January 2009.

[8] T. Ju, Q.-Y. Zhou, and S.-M. Hu. Editing the topology of 3d models by sketching. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 42, New York, NY, USA, 2007. ACM.

[9] J. Lachaud and A. Montavert. Deformable meshes with automated topology changes for coarse to fine 3d surface extraction. *Medical Image Analysis*, 3(2), 1999.

[10] J. Landay and B. Myers. Sketching interfaces: toward more human interface design. *Computer*, 34(3):56 –64, mar 2001.

[11] D. Lesage, E. D. Angelini, I. Bloch, , and G. Funka-Lean. Bayesian maximal paths for coronary artery segmentation from 3d ct angiograms. In *Medical Image Computing and Computer-Assisted Intervention ï¿½ MICCAI 2009*, pages 222–229, 2009.

[12] J. Maintz and M. A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1 − 36, 1998.

[13] M. Masry and H. Lipson. A sketch-based interface for iterative design and analysis of 3d objects. In *ACM SIGGRAPH 2007 courses*, SIGGRAPH '07, New York, NY, USA, 2007. ACM.

[14] A. Nealen, O. Sorkine, M. Alexa, and D. Cohen-Or. A sketch-based interface for detail-preserving mesh editing. *ACM Trans. Graph.*, 24:1142–1147, July 2005.

[15] L. Olsen, F. F. Samavati, M. C. Sousa, and J. A. Jorge. Sketch-based modeling: A survey. *Computers & Graphics*, 33(1):85 − 103, 2009.

[16] A. Pihuit, M.-P. Cani, and O. Palombi. Sketch-based modeling of vascular systems: a first step towards interactive teaching of anatomy. In *SBIM 2010, June, 2010*, pages 151–158, Annecy, France, June 2010.

[17] A. Sharf, T. Lewiner, A. Shamir, and L. Kobbelt. On-the-fly curve-skeleton computation for 3d shapes. *Comput. Graph. Forum*, 26(3):323–328, 2007.

[18] A. Sharf, T. Lewiner, G. Shklarski, S. Toledo, and D. Cohen-Or. Interactive topology-aware surface reconstruction. *ACM Trans. Graph.*, 26(3):43, 2007.

[19] A. Tagliasacchi, H. Zhang, and D. Cohen-Or. Curve skeleton extraction from incomplete point cloud. *ACM Trans. Graph.*, 28:71:1–71:9, July 2009.

[20] A. Wilson and A. Bobick. Nonlinear phmms for the interpretation of parameterized gesture. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:879, 1998.

[21] J. Wither, F. Bertails, and M.-P. Cani. Realistic hair from a sketch, June 2007.

[22] J. O. Wobbrock, A. D. Wilson, and Y. Li. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*, UIST '07, pages 159–168, New York, NY, USA, 2007. ACM.

[23] C.-Y. Yao, H.-K. Chu, T. Ju, and T.-Y. Lee. Compatible quadrangulation by sketching. *Comput. Animat. Virtual Worlds*, 20:101–109, June 2009.

[24] A. Yureidini, E. Kerrien, and S. Cotin. Reconstruction robuste des vaisseaux sanguins par surfaces implicites locales. In *Orasis*, Praz-sur-Arly, France, June 2011.

Contents
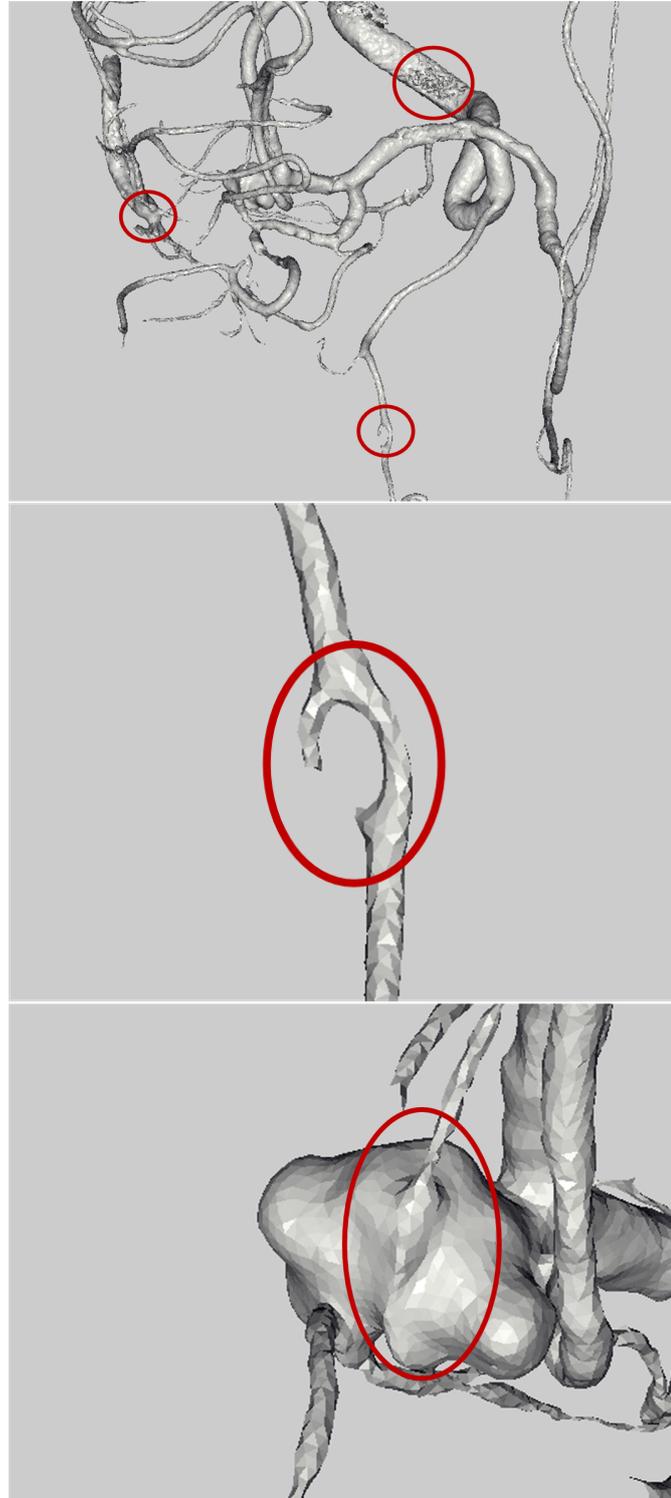
Figure 1: Top: result of an automatic reconstruction of brain vascular system, from medical images. Middle and bottom, two examples of classical artifacts induced by such automatic approach: topological aberration on a small artery, and a small artery within an aneurysm (also known as a kissing vessel artifact), respectively.
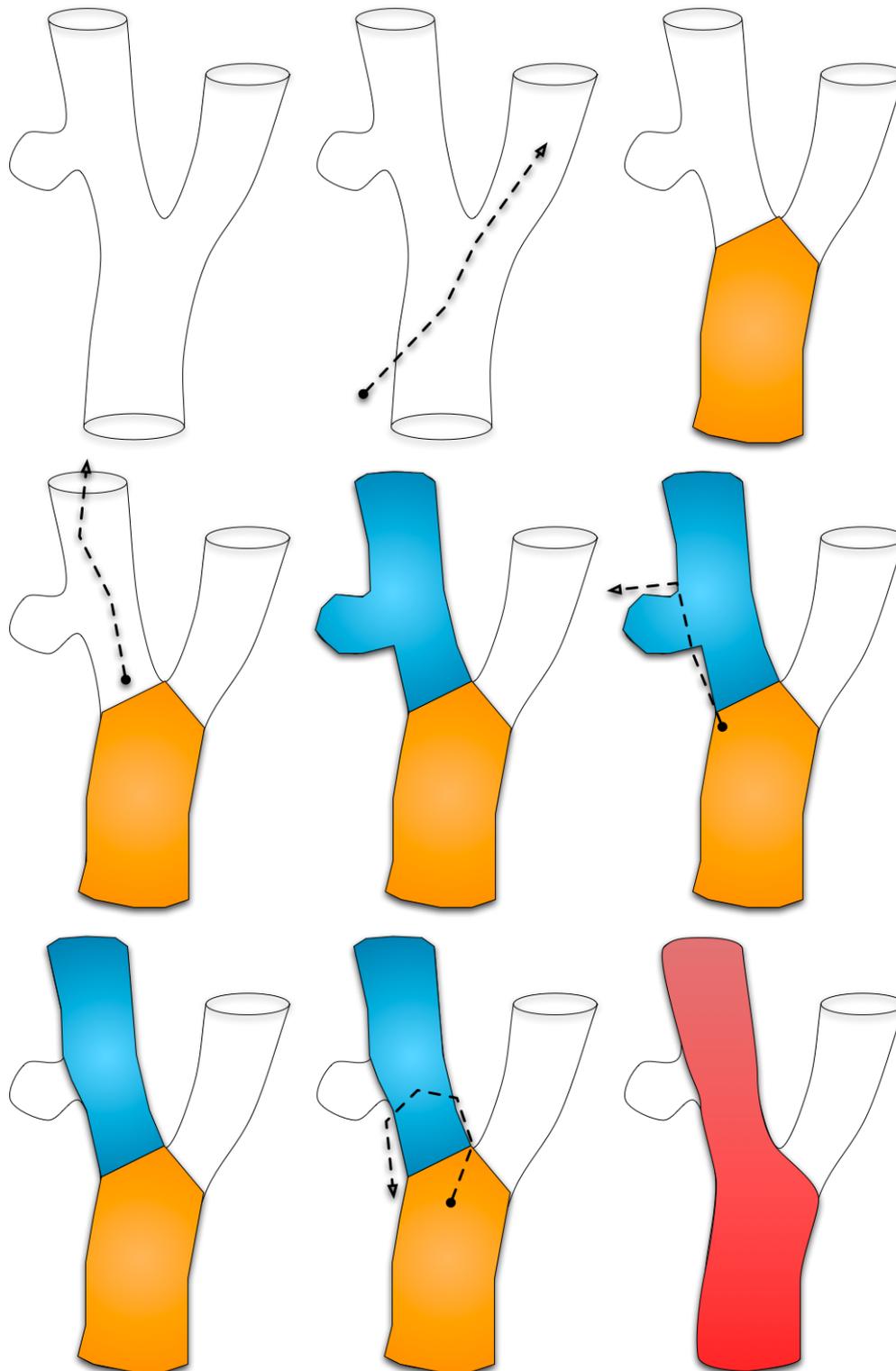
Figure 2: Overview of the approach: given a geometry, the user can perform 3 different gestures to tag the topology: linear stroke, circular stroke, walking stick stroke. This 3 tools allow to easily tag the geometry of the triangular mesh.
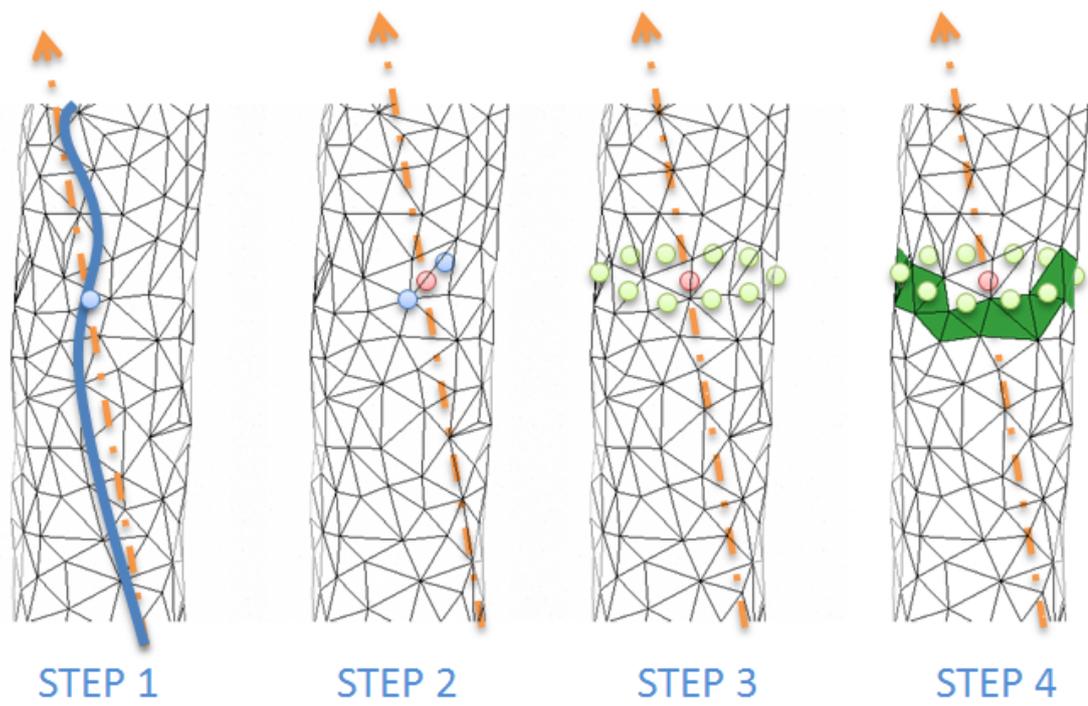
Figure 3: Steps to set up the propagation seed
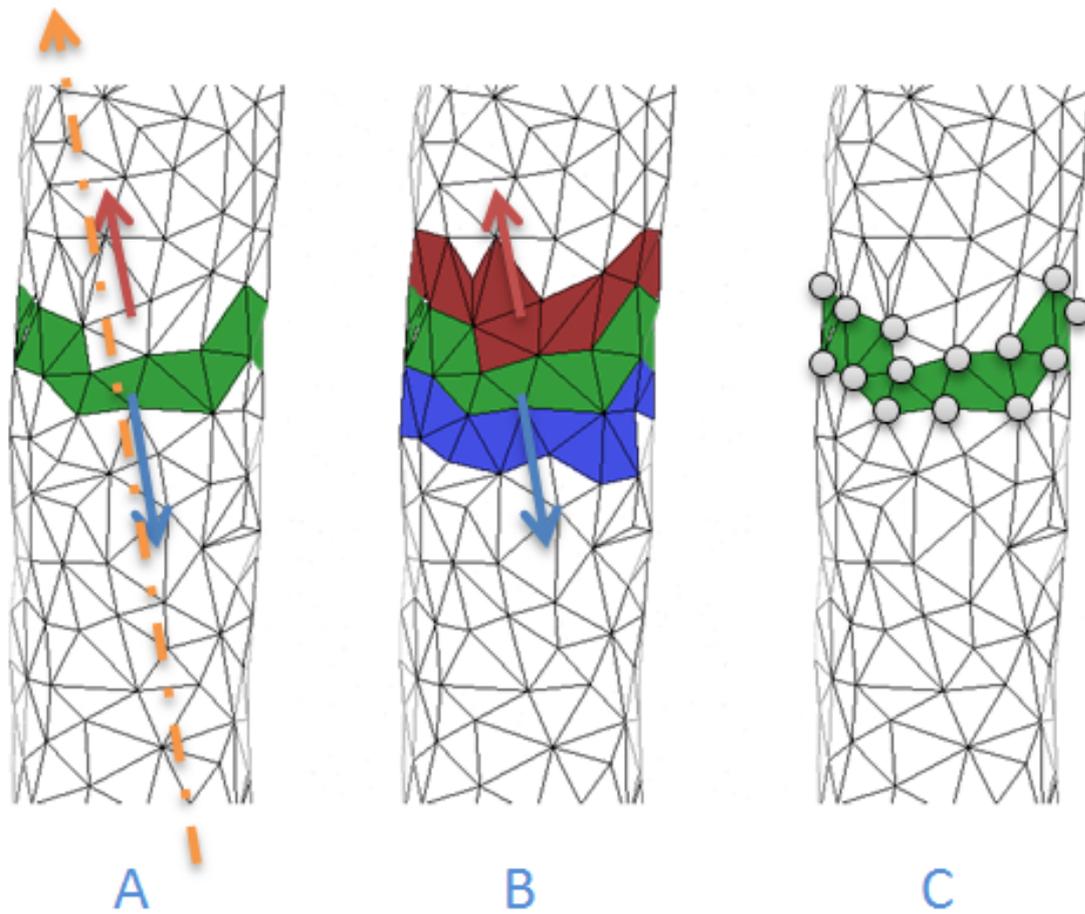
Figure 4: The second step of progation after the seed creation to get closest rings, in orginal direction (red) and opposite direction (blue)
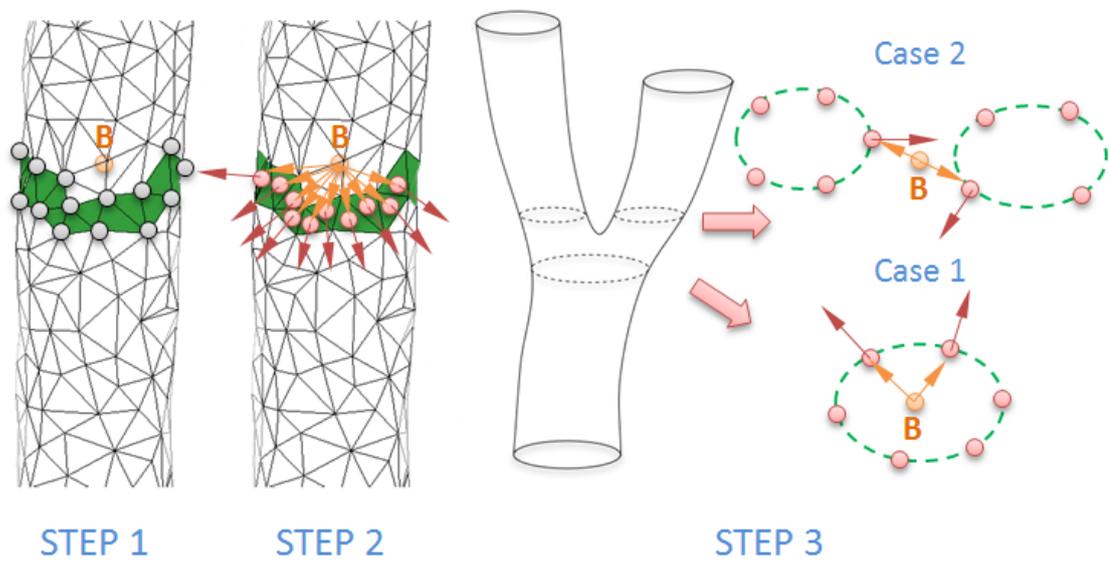
Figure 5: Stop criterion: between two steps of the propagation, and according to the scalar defined in each point of current ring the algorithm can found two cases: the ring's average center is in the ring (below right) and average center is out (upper right).
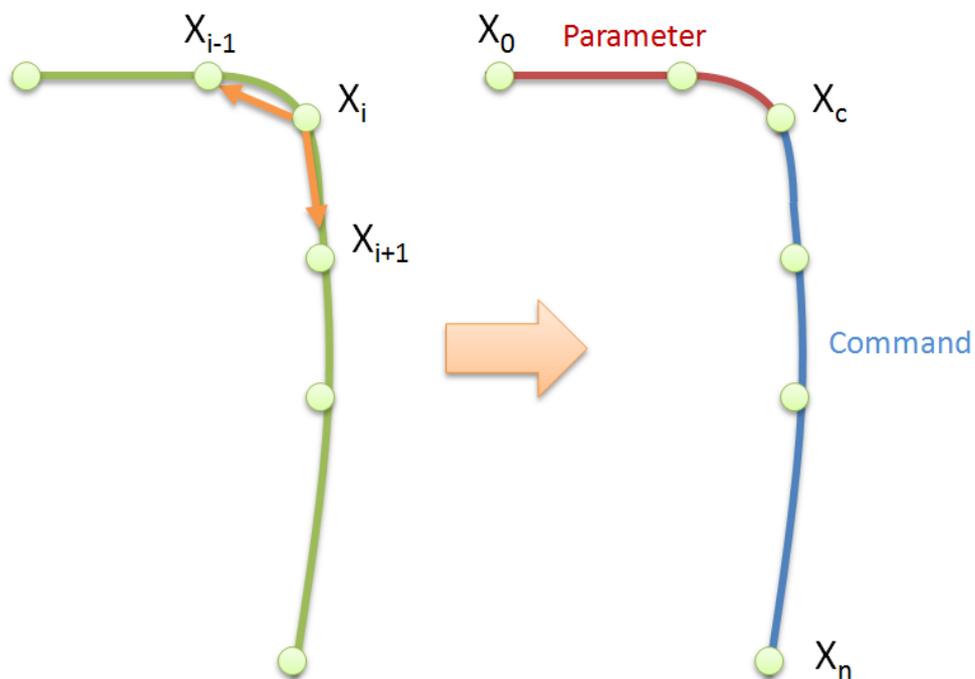
Figure 6: After recognition of the user's input (on the left) as a walking stick stroke, the algorithm determines which parts are respectively the command and the parameter (on the right). This separation is defined by the curvature maximum point of user gesture.
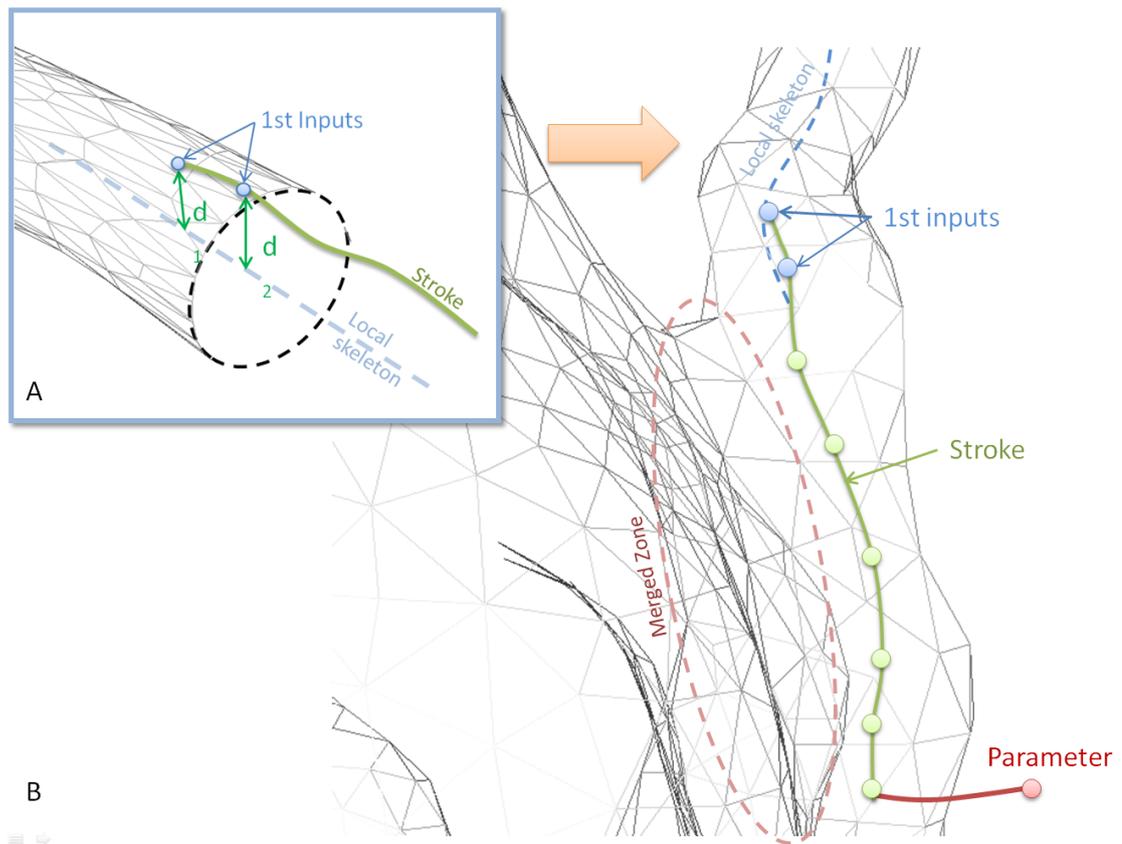
Figure 7: Application of the walking stick stroke algorithm on merged zone: the user begins his gesture on a vessel already tagged (picture A), and continues his gesture by drawing the path of the kissing vessel (picture B). This action topologically disjoins the two structures.